WOLFRAM RESEARCH mathworld.wolfram.com

ASTRONOMY BIOGRAPHY CHEMISTRY MATHEMATICS PHYSICS Search Site

ERIC WEISSTEIN'S world of MATHEMATICS

INDEX BY SUBJECT
- Algebra
- Applied Mathematics
- Calculus and Analysis
- Discrete Mathematics
- Foundations of Mathematics
- Geometry
- History and Terminology
- Number Theory
- Probability and Statistics
- Recreational Mathematics
- Topology

ALPHABETICAL INDEX

- ABOUT THIS SITE
- AUTHOR'S NOTE
- FAQs
- WHAT'S NEW
- RANDOM ENTRY
- BE A CONTRIBUTOR
- SIGN THE GUESTBOOK
- EMAIL COMMENTS
- HOW CAN I HELP?
- TERMS OF USE

ERIC'S OTHER SITES

ORDER BOOK FROM AMAZON.COM

Probability and Statistics ▸ Probability ▾

# Marginal Probability

Let $S$ be partitioned into $r \times s$ disjoint sets $E_i$ and $F_j$ where the general subset is denoted $E_i \cap F_j$. Then the marginal probability of $E_i$ is

$$P(E_i) = \sum_{j=1}^{s} P(E_i \cap F_j).$$

SEE ALSO: Conditional Probability, Distribution Function, Joint Distribution Function, Probability Function

*Author: Eric W. Weisstein*

WOLFRAMRESEARCH mathworld.wolfram.com

ASTRONOMY BIOGRAPHY CHEMISTRY MATHEMATICS PHYSICS Search Site

ERIC WEISSTEIN'S world of MATHEMATICS

INDEX BY SUBJECT
- Algebra
- Applied Mathematics
- Calculus and Analysis
- Discrete Mathematics
- Foundations of Mathematics
- Geometry
- History and Terminology
- Number Theory
- Probability and Statistics
- Recreational Mathematics
- Topology

ALPHABETICAL INDEX

- ABOUT THIS SITE
- AUTHOR'S NOTE
- FAQs
- WHAT'S NEW
- RANDOM ENTRY
- BE A CONTRIBUTOR
- SIGN THE GUESTBOOK
- EMAIL COMMENTS
- HOW CAN I HELP?
- TERMS OF USE

ERIC'S OTHER SITES

ORDER BOOK FROM AMAZON.COM

Probability and Statistics ▸ Statistical Distributions ▸ General Distributions ▾

# Statistical Distribution

The distribution of a variable is a description of the relative numbers of times each possible outcome will occur in a number of trials. The function describing the distribution is called the probability function, and the function describing the cumulative probability that a given value *or any value smaller than it* will occur is called the distribution function.

Formally, a distribution can be defined as a normalized measure, and the distribution of a random variable $x$ is the measure $P_x$ on $\mathbb{S}'$ defined by setting

$$P_x(A') = P\{s \in S : x(s) \in A'\},$$

where $(S, \mathbb{S}, P)$ is a probability space, $(S, \mathbb{S})$ is a measurable space, and $P$ a measure on $\mathbb{S}$ with $P(S) = 1$. If the measure is a Radon measure (which is usually the case), then the statistical distribution is a generalized function in the sense of a generalized function.

SEE ALSO: Continuous Distribution, Discrete Distribution, Distribution Function, Generalized Function, Measurable Space, Measure, Probability, Probability Density Function, Random Variable, Statistics

## References

Doob, J. L. "The Development of Rigor in Mathematical Probability (1900-1950)." *Amer. Math. Monthly* **103**, 586-595, 1996.

Evans, M.; Hastings, N.; and Peacock, B. *Statistical Distributions, 3rd ed.* New York: Wiley, 2000.

*Author: Eric W. Weisstein*

S3.1

# FAST SPEAKER ADAPTATION FOR SPEECH RECOGNITION SYSTEMS

F. Class (1) A. Kaltenmeier (1) P. Regel (1) K. Trottler (2)
(1) Daimler Benz AG Research Institute, Ulm, FRG
(2) TelefunkenSystemTechnik, Deutsche Aerospace, Ulm, FRG

## Abstract

This paper deals with different speaker adaptation methods for speech recognition systems adapting automatically to new and unknown speakers in a short training phase. The adaptation techniques aim at transformations of feature vectors, optimized with respect to some constraints. Two different adaptation strategies are appropriate. The first one is based on least mean squared error (MSE) optimization. The second method is a codebook-driven feature transformation. Both adaptation techniques are incorporated into two different recognition systems: dynamic time warping (DTW) and Hidden Markov Modelling (HMM). The results show, that in both systems speaker-adaptive error rates are close to speaker-dependent error rates. In the best case the mean error rate of four test speakers decreases by a factor of 6 (DTW-recognizer) resp. 3 (HMM-recogniser) compared to the inter-speaker error rate without adaptation. Finally a hardware realisation of the speaker-adaptive HMM-recogniser will be described.

## 1 Introduction

Fast speaker adaptation is of increasing importance for speech recognition systems with large vocabulary. The traditional way to train a system to each user's voice (speaker-dependent system) is to utter each vocabulary word once or several times. This procedure is no longer acceptable with increasing vocabulary size. Furthermore recognition schemes such as HMM or Neural Networks need a high amount of training data to optimize the classification parameters.
Hence new methods are applied to adapt the system to a new speaker. The recognition system is pretrained using training data of one or several reference speakers. This primary training effort may be very high. The system is then adapted to a new unknown user, who has to utter only few words or phrases. Two strategies have been investigated during the last years: adaptation of the pretrained classification parameters [1-3] and adaptation by transformation of the feature vectors [4-7]. We pursue the second strategie called *spectral mapping*, where the problem is to find a suitable transformation.
In section 2 several optimisation methods are described. These methods are tested with two different classification schemes: DTW and HMM. Experiments and results are shown in section 3. Finally a description of a hardware realisation is given in section 4.

## 2 Feature Vector Transformations

We have investigated five methods in order to determine a suitable transformation. Four of them use transformation matrices optimised with respect to the MSE-criterion. The fifth method is an adaptation strategy based on vector quantisation.

### 2.1 Common MSE-Optimization

Let us find a transformation $\mathbf{x} = \mathcal{A}^T \cdot \mathbf{X}$, where $\mathbf{X}$ is a feature vector of the new speaker, $\mathcal{A}$ is a transformation matrix and $\mathbf{x}$ is the transformed vector. To estimate $\mathcal{A}$ we minimise the mean-squared error

$$D = E\left[\left(\mathbf{Y} - \mathcal{A}^T \cdot \mathbf{X}\right)^T \cdot \left(\mathbf{Y} - \mathcal{A}^T \cdot \mathbf{X}\right)\right] \qquad (1)$$

between a feature vector $\mathbf{Y}$ of the reference speaker and the corresponding transformed vector $\mathcal{A}^T \cdot \mathbf{X}$ of the new speaker. The solution of this minimisation problem is given by

$$\mathcal{A} = \left(E\left[\mathbf{X}\mathbf{X}^T\right]\right)^{-1} \cdot E\left[\mathbf{X}\mathbf{Y}^T\right]. \qquad (2)$$

The expected values $E\left[\mathbf{X}\mathbf{X}^T\right]$ and $E\left[\mathbf{X}\mathbf{Y}^T\right]$ are estimated using *corresponding* vectors $\mathbf{X}$ and $\mathbf{Y}$ of the same words uttered by the new and reference speaker in a short training phase. In order to get a proper time-alignment we use DTW without slope constraint, i.e. each reference utterance $\{\mathbf{Y}\}$ is time-warped against each corresponding utterance $\{\mathbf{X}\}$ of the new speaker yielding a set of corresponding vectors $\{\mathbf{X}(i), \mathbf{Y}(i)\}$, where i are frame indices. This procedure is used identically with all methods described in this paper.
The optimisation strategy works well for mel-frequency-coefficients (MFC), but it denies for mel-cepstral-coefficients (MCC), i.e. the recognition rate decreases if the transformation is applied. Evidently additional constraints are necessary to find an optimal transformation for MCCs.

### 2.2 MSE-Optimization with constraints

The method is called MSE_C. In the following we introduce the constraint, that the variance of each component $x_k$ of the *transformed* vector $\mathbf{x}$ should be equal to the variance of the corresponding component of the reference vector $\mathbf{Y}$

$$E\left[x_k^2\right] = E\left[Y_k^2\right]. \qquad (3)$$

With ( 1) we get a new optimization criterion written for each component k

$$D_k = E\left[x_k^2\right] - 2E\left[x_k \cdot Y_k\right] + E\left[Y_k^2\right] \overset{!}{=} min. \qquad (4)$$

Obviously the error component $D_k$ is minimized, if the correlation

$$E\left[x_k \cdot Y_k\right] = \mathbf{a}_k^T \cdot E\left[\mathbf{X}Y_k\right] \overset{!}{=} max \qquad (5)$$

*This work was partly supported by the German Federal Research and Technology Departement Agency (BMFT) under grant No. ITM 8801 D. Only the authors are responsible for the contents of this publication.

CH2847-2/90/0000-0133 $1.00 © 1990 IEEE

between the transformed vector component $x_k = \mathbf{a}_k^T \cdot \mathbf{X}$ and the reference vector component $Y_k$ is maximised, where $\mathbf{a}_k$ is a transformation vector. To solve this maximization problem under the constraint ( 3) we use the Lagrange method and obtain

$$\mathbf{a}_k = \frac{1}{\lambda_k} \left(E\left[\mathbf{X}\mathbf{X}^T\right]\right)^{-1} \cdot E\left[\mathbf{X}Y_k\right] \qquad (6)$$

with the Lagrange parameter

$$\lambda_k^2 = \frac{1}{E\left[Y_k^2\right]} \cdot E\left[\mathbf{X}Y_k\right]^T \cdot \left(E\left[\mathbf{X}\mathbf{X}^T\right]\right)^{-1} \cdot E\left[\mathbf{X}Y_k\right]. \qquad (7)$$

Eq. ( 6) corresponds - with the exception of the factor $1/\lambda_k$ - to the solution ( 2) in the unconstraint case. Finally the desired matrix $\mathcal{A}$ is built up with the vector $\mathbf{a}_k$ as the k-th column vector.

### 2.3 Transformation into a joint feature space

This method (called **GRE**) is based on a technique for spectral transformation proposed by GRENIER et al.[5]. The idea is to transform the feature vectors $\mathbf{X}$ of the new speaker as well as the vectors $\mathbf{Y}$ of the reference speaker into a joint feature space by means of linear transformations $\mathbf{x} = \mathcal{P}_L \cdot \mathbf{X}$, $\mathbf{y} = \mathcal{P}_R \cdot \mathbf{Y}$. The transformation matrices $\mathcal{P}_L$ and $\mathcal{P}_R$ have to be determined in such a way, that the Euclidian metric

$$D = E\left[(\mathcal{P}_L \cdot \mathbf{X} - \mathcal{P}_R \cdot \mathbf{Y})^T (\mathcal{P}_L \cdot \mathbf{X} - \mathcal{P}_R \cdot \mathbf{Y})\right] \qquad (8)$$

between the *transformed* vectors $\mathbf{x}$ and $\mathbf{y}$ is minimum. Since the trivial, but non satisfactory solution $\mathcal{P}_R = \mathcal{P}_L = 0$ accomplishes this criterion, we introduce the additional constraint

$$E\left[x_k^2\right] = E\left[y_k^2\right] = 1, \qquad (9)$$

i.e. we require unit variance for the components of both transformed vectors. With respect to this normalisation the minimisation problem can now be formulated for each component according to

$$D_k = E\left[(x_k - y_k)^2\right] = 2\left(1 - E\left[x_k y_k\right]\right) \overset{!}{=} min. \qquad (10)$$

Thus $D_k$ is minimum, if the components $x_k$ and $y_k$ of the target vectors $\mathbf{x}$ and $\mathbf{y}$ are maximally correlated. The solution can be found by applying 'canonical correlation analysis' [7,8]. This leads to a generalised eigenproblem, which can be solved by employing techniques known from the singular value decomposition [9] of a matrix. This matrix contains the auto- and crosscovariance matrices of the two speakers, which are estimated using corresponding feature vectors in a short training phase.

A modified version (**GRE_1T**) of this method is to combine the two matrices $P_L$ and $P_R$ to a single one in order to avoid computations for the speaker adaptation during application, i.e. only the feature vectors $\mathbf{Y}$ of the reference speaker are transformed once after the short training phase:

$$\hat{\mathbf{y}} = P_L^{-T} \cdot P_R^T \cdot \mathbf{Y}(i) = \left(P_R \cdot P_L^{-1}\right)^T \cdot \mathbf{Y}(i) \qquad (11)$$

### 2.4 Nonlinearly extended feature vectors

It is obvious, that the performance of an adaptation procedure with linearly transformed vectors is limited due to nonlinear dependencies. On the other hand optimization problems based on linear transformations can be solved in closed forms. We can combine these two reflections by applying the linear transformations to nonlinearly extended feature vectors (**GRE_Q**). A primary feature vector $\mathbf{v} = (v_1, v_2, \cdots, v_K)$ is extended here to a polynomial vector of second order by forming quadratic combinations of the components: $\mathbf{v}_Q = (v_1, v_2, \cdots, v_K, v_1^2, v_1 v_2, \cdots, v_K^2)$. This extension is performed for the test as well as for the reference templates. Concerning the calculation of the transformation matrices we can proceed in the same way as described above.
A combination of the two matrices to one is possible, too. This method is called **GRE_Q_1T**.

### 2.5 Transformation by use of a codebook

The idea is to use a quantised feature space in order to get a suitable transformation. This is done by means of a codebook. Thus any nonlinear transformation can be realised. Each feature vector of the reference speaker is mapped into the quantised feature space through vector quantisation yielding codebook symbol $S_m$ and then replaced by a new feature vector, which is related to this Voronoi cell $S_m$ (note: the new feature vector is not the centroid of the cell). This new feature vector has been created by a linear combination of feature vectors of the new speaker (the method may be applied vice versa, too). For computing the linear combinations we use a codebook. After investigating several variations, the following procedure (method **CB**, fig. 1) was implemented:

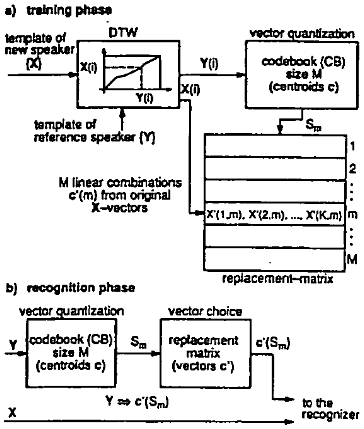
a) training phase
template of new speaker (X)
DTW
X(i)
Y(i)
Y(i)
X(i)
vector quantization
codebook (CB) size M (centroids c)
template of reference speaker (Y)
$S_m$
1
2
M linear combinations c'(m) from original X-vectors
X'(1,m), X'(2,m), ..., X'(K,m)
m
M
replacement-matrix
b) recognition phase
vector quantization
Y
codebook (CB) size M (centroids c)
$S_m$
vector choice
replacement matrix (vectors c')
c'($S_m$)
Y ⇒ c'($S_m$)
X
to the recognizer

Figure 1: Implementation of the CB-adaptation-strategy

- Corresponding vectors $\mathbf{X}(i)$ and $\mathbf{Y}(i)$ are determined using DTW.
- Each reference vector $\mathbf{Y}(i)$ is uniquely mapped into a codebook symbol $S_m$ using vector quantization.
- For each codebook symbol $S_m$ we now compute the mean vector $\mathbf{c}(\mathbf{m})$ of all vectors $\mathbf{X}(i)$, whose *corresponding* vectors $\mathbf{Y}(i)$ mapped into $S_m$. This mean vector can be computed recursively during the training phase.
- At the end of the training phase each reference vector $\mathbf{Y}(i)$ is replaced by the linear combination $\mathbf{c}(m)$, which corresponds to its code symbol $S_m$.

In the recognition phase the vectors $\mathbf{X}(i)$ of the new speaker remain unchanged. This adaptation strategy is similar to that one proposed by Shikano et al. [4].

## 3 Experiments and Results

We used two different recognition schemes in order to test the performance of the various adaptation methods: a DTW- and HMM-recogniser.

**Test conditions:** The signal was low-pass filtered and sampled with 12 kHz sampling frequency. Then mel-cepstral-coefficients (MCC) were computed every 10 msec. We used $K = 10$ MCCs per feature vector. Speaker-dependent mean values of the MCC's were subtracted. The test-vocabulary consisted of 100 common german words, spoken by 4 male (A;B;C;D) and 1 female (E) speaker. Two sets of 100 words/speaker (S1 and S2) were recorded on two different days. Furthermore a database of about 15 min. of speech was established, spoken by speaker A in order to design a codebook and to train the HMM's. Therefore speaker A was defined as reference speaker; the remaining 4 speakers formed the test set.

The quality of the adaptation can be evaluated by comparing the speaker-dependent error rates (SD), the speaker-adaptive error rate and the error rate without any adaptation (WA). The SD was measured using S1 and S2 of a speaker as reference resp. test set (DTW-recogniser). WA is obtained by classifying S2 of all speakers except ref. speaker A. The adaptation of speaker B to A e.g., is performed using S1 of both speakers to compute transformation matrices. The performance of the adaptation is controlled by classifying S2 of each test speaker after transformation of the according speech samples.

**DTW-recognizer:** The DTW-system was an isolated word recognizer based on *city-block* distance measure, which was not optimized for extraordinarily high performance: Fig. 2 and table 1 show the results for all test speakers, each of them adapted to reference speaker A.

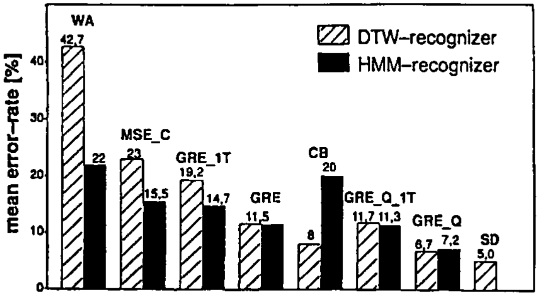

Figure 2: Mean error rates [%] of the different adaptation methods; 100 words in training phase (≈ 1.5 min. of speech)

The mean error rate decreases from 42.7% without adaptation to 23% (MSE_C), 19.2% (GRE_1T), 11.5% (GRE), 8% (CB), 6.7% (GRE_Q). For method CB we used a speaker-dependent codebook (size 256). The result of method CB with speaker-*independent* codebook (10 extraneous speakers, 900 word vocabulary, size 256) was 10%.

The best results have been achieved by applying GRE_Q. The error rate is by a factor of 6 below WA and is within the scope of the SD (5%). It is worth noting, that the best results are obtained by using only the first 10 *transformed* components of the quadratically extended feature vectors for further classification.

Speaker adaptation should aim at a short training phase. Therefore the amount of samples necessary for optimizing the transformation matrices is another criterion for evaluating the performance of the adaptation procedure. Thus we used the first n ($10 \leq n \leq 100$) templates of the training sets. From fig. 3 it becomes obvious that for the GRE-method 20 words are sufficient to train the required parameters. The other methods, however, need about 40 words for convergence.

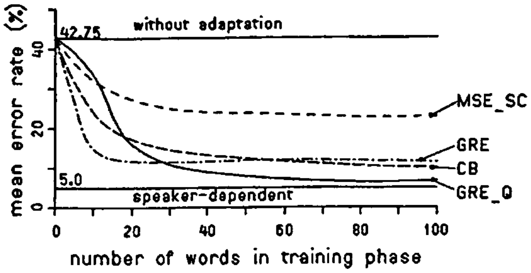

Figure 3: Mean error rate vs. different number of training templates; DTW-recognizer

A further reduction of the training phase with respect to the new speaker is possible if we store several repetitions of the *reference* speaker's utterances. Tested with CB-method we obtained about the same results for a training vocabulary spoken once by new and reference speaker compared to a training vocabulary of *half* size spoken once by the new speaker and 5 times by the reference speaker.

The influence of the reference speaker was investigated in a further experiment, i.e. we used speaker B instead of speaker A as the reference speaker. The results differ only slightly from those above for all methods. Therefore we conclude, that the choice of the reference speaker is no critical parameter for the adaptation methods. However, further investigations will be necessary to confirm these results.

Another point of view was the choice of the adaptation vocabulary. Some experiments showed that the training phase can be minimised if the training vocabulary is phonetically balanced. If not, the training phase must be longer to get optimal results.

**HMM-recognizer:** Our HMM recognition system is described in detail in [6]. Words are represented by a series of HMM's of subword units. The phonetic graphs, whose nodes are the HMM's, are automatically generated by rules from the standard orthographic descriptions of words. These descriptions are stored in the Lexicon. The phonetic description of a word is a graph because usual alternate pronounciations are taken into account using these rules.

HMM's are described by continuous transition and discrete emission probabilities. Therefore vector quantization (VQ) is necessary. VQ is carried out by means of a *speaker-dependent* codebook (size 128). Furthermore we use *speaker-dependent* models of subword units.

Applying the proposed adaptation methods we have to incorporate feature vector transformations in the HMM-system. For this purpose transformation matrices are computed in the same manner as described above.

For the application we have to distinct between *one-sided* and *two-sided* adaptations (transformation of one or both speaker's vectors). The one-sided methods MSE_C, GRE_1T, GRE_Q_1T and CB are directly applicable to transform the

|  | WA |  | MSE_C |  | GRE_1T |  | GRE |  | GRE_Q_1T |  | GRE_Q |  | CB |  | SD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | DTW | HMM | DTW | HMM | DTW | HMM | DTW | HMM | DTW | HMM | DTW | HMM | DTW | HMM | DTW |
| B | 32 | 30 | 18 | 15 | 20 | 19 | 11 | 15 | 13 | 16 | 9 | 11 | 8 | 24 | 7 |
| C | 20 | 20 | 15 | 17 | 12 | 14 | 7 | 10 | 16 | 11 | 4 | 6 | 9 | 19 | 5 |
| D | 67 | 21 | 34 | 11 | 27 | 17 | 12 | 9 | 8 | 11 | 5 | 2 | 4 | 22 | 2 |
| E | 52 | 17 | 25 | 19 | 18 | 15 | 16 | 12 | 10 | 12 | 9 | 10 | 11 | 15 | 6 |
| $\mu$ | 42.7 | 22 | 23 | 15.5 | 19.2 | 14.7 | 11.5 | 11.5 | 11.7 | 11.3 | 6.7 | 7.2 | 8 | 20 | 5 |

Table 1: Error rates [%] of all adaptation methods; ref. speaker A, test speakers B-E; training at 100 words.

feature vectors of the new speaker. On the opposite, the two-sided methods GRE and GRE_Q transform the vectors of the reference speaker, too. In a real application we don't have application vocabulary feature vectors of the reference speaker, i.e. he is represented by the codebook and HMM's are trained using this codebook. Therefore the conventional but expensive way for speaker adaptation would be the following: transformation of all training material (reference speaker's utterances); codebook-generation with transformed data; training the HMM's with transformed data. This procedure can be shortened by transforming the codebook centroids instead of reference speaker's vectors. This is equivalent to a transformation of the *quantized* reference speaker's space. Hence the one-sided adaptation methods are applicable in the same manner, i.e. transforming the codebook centroids instead of the new speaker's feature vectors to adapt the reference speaker (the system) to the new speaker.
The results of the speaker adaptive HMM-system are shown in figure 2 and table 1.

In principle there is the same behaviour of both DTW- and HMM-recognizer. Method GRE_Q is the best, too, with a mean error rate of 7.2%. This is by a factor of 3 below WA. We obtained significant reduction of the mean error rate for all adaptation methods with the exception of CB-method. Giving excellent results for DTW-recognizer, it is only slightly better than WA of the HMM-recognizer. We will try to find out the reason in our future work.

## 4 Realization of the speaker-adaptive HMM-recognizer

The algorithms of the speaker-adaptive HMM-recognizer are implemented on a PC-based demonstration system. Data acquisition (MCC-computation) as well as preprocessing are realised on a purchasable PC-board on the basis of the TMS 320C25.
A second board is based on the floating-point signal processor DSP32C from AT&T and appropriate for performing speaker adaptation and classification. A floating- point processor was chosen because of algorithmic complexity. The feature vectors are transformed and quantized using the codebook of the reference speaker. Finally the HMM-algorithm based on subword models determines the N best word candidates, the labels of which are transmitted to the host.
The DSP32C board is equipped with a memory extension board of 4 MByte. At the moment the system is designed to operate with an active vocabulary of about 1000 isolated words. The concept, however, is kept flexible. Therefore a larger vocabulary as well as a continuous speech recognition option can be integrated easily.

## 5 Conclusions

We have investigated several speaker adaptation methods by feature vector transformations. Most of the proposed methods use only one transformation matrix (one-sided adaptation), i.e. they can be organized in such a way, that the refeence speaker's vectors are transformed once after the training phase and therefore no computations are necessary for adaptation in the application phase. Two methods are two-sided, which transform both the new and ref. speaker's vectors.
The adaptation procedure consists of two steps: a) computing a transformation matrix (resp. matrices) automatically using a few utterances (20 - 40) spoken by the new speaker in a short training phase, and b) transforming the feature vectors. The methods are invariant with respect to vocabulary and classification scheme because they are based on unlabeled feature vectors. The experiments indicate that all methods result in significant improvements, some of which lie in the scope of the speaker-dependent error rate. In the best case the mean error rate decreases by a factor of 6 (DTW-recogniser) resp. 3 (HMM-recogniser) compared to the inter-speaker error rate without adaptation.

### References

[1] H. Bonneau et al.: Vector Quantization for Speaker Adaptation. ICASSP87, Dallas, pp. 1434-1437.
[2] M. Nishimura et al.: Speaker Adaptation Method for HHM-Based Speech Recognition. ICASSP88, New York, pp. 207-210.
[3] M. Feng et al.: Iterative Normalization for Speaker-Adaptive Training in Continuous Speech Recognition. ICASSP89, Glasgow, pp. 612-615.
[4] K. Shikano et al. : Speaker Adaptation Through Vector Quantization. ICASSP86, Tokyo, pp. 2643-2646.
[5] K.CHOUKRI, G.CHOLLET, Y.GRENIER: Spectral Transformations through Canonical Correlation Analysis for Speaker Adaptation in ASR. ICASSP86, Tokyo, pp. 2659-2662.
[6] F.Class, A.Kaltenmeier, P.Regel: Speaker Adaptive Word Verification Using Hidden Markov Models of Sound Units for a Recognition System with large Vocabulary. Proc. of 7th FASE Symposium SPEECH 88, Edinburgh, pp. 23-30.
[7] F. Class et al.:Speaker Adaptation for Recognition Systems with a Large Vocabulary. Proc. of MELECON 89, April 1989, Lissabon, pp. 241-244.
[8] T.W. ANDERSON: An Introduction to Multivariate Statistical Analysis. J.Wiley & Sons, New York, 1958.
[9] G.H.Golub, C.Reinsch: Singular value decomposition and least squares solutions. Numerische Mathematik, vol. 14, 1970, pp.403-420.